11-28-2007

The Exponential Function of Matrices

Nathalie Nicholle Smalls

THE EXPONENTIAL FUNCTION OF MATRICES

by

NATHALIE SMALLS

Under the Direction of Marina Arav

ABSTRACT

The matrix exponential is a very important subclass of functions of matrices that has been studied extensively in the last 50 years. In this thesis, we discuss some of the more common matrix functions and their general properties, and we specifically explore the matrix exponential. In principle, the matrix exponential could be computed in many ways. In practice, some of the methods are preferable to others, but none are completely satisfactory. Computations of the matrix exponential using Taylor Series, Scaling and Squaring, Eigenvectors, and the Schur Decomposition methods are provided.

Keywords: Matrix Exponential, Functions of Matrices, Jordan Canonical Form, Matrix Theory, Scaling and Squaring, Schur Decomposition

THE EXPONENTIAL FUNCTION OF MATRICES

by

NATHALIE SMALLS

A Thesis Presented in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in College of Arts and Sciences

Georgia State University

2007

THE EXPONENTIAL FUNCTION OF MATRICES

by

NATHALIE SMALLS

Major Professor:	Marina Arav
Committee:	Rachel Belinsky
	Frank Hall
	Zhongshan Li
	Michael Stewart

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December 2007

ACKNOWLEDGMENTS

The author wishes to gratefully acknowledge the assistance of Drs. Marina Arav, Rachel Belinsky, Frank J. Hall, Zhongshan Li, and Michael Stewart without whose guidance this thesis would not have been possible. She would also like to thank Drs. Mihaly Bakonyi, Lifeng Ding, Florian Enescu, Pulak Ghosh, Yu-Sheng Hsu, Gengsheng Qin, Draga Vidakovic, and Joseph J. Walker for their support and encouragement in her course work and in her research towards this thesis.

Contents

1 INTRODUCTION

As stated in [1] and [19], the introduction and development of the notion of a matrix and the subject of linear algebra followed the development of determinants. Gottfried Leibnitz, one of the two founders of calculus, used determinants in 1693 arising from the study of coefficients of systems of linear equations. Additionally, Cramer presented his determinant-based formula, known as Cramer's Rule, for solving systems of linear equations in 1750. However, the first implicit use of matrices occurred in Lagrange's work on bilinear forms in the late 1700's in his method now known as Lagrange's multipliers. Some research indicates that the concept of a determinant first appeared between 300 BC and 200 AD, almost 2000 years before its invention by Leibnitz, in the Nine Chapters of the Mathematical Art by Chiu Chang Suan Shu. There is no debate that in 1848 J.J. Sylvester coined the term, "matrix", which is the Latin word for womb, as a name for an array of numbers. Matrix algebra was nurtured by the work of Issai Schur in 1901. As a student of Frobenius, he worked on group representations (the subject with which he is most closely associated), but also in combinatorics and even theoretical physics. He is perhaps best known today for his result on the existence of the Schur decomposition, which is presented later in this thesis.

In mathematics, a matrix is a rectangular table of numbers, or more generally, a table consisting of abstract quantities. Matrices are used to describe linear equations, keep track of coefficients of linear transformations, and to record data that depend on two parameters. Matrices can be added, multiplied, and decomposed in various ways, which makes them a key concept in linear algebra and matrix theory, two of the fundamental tools in mathematical disciplines. This makes intermediate facts about matrices necessary to understand nearly every area of mathematical science, including but not limited to differential equations, probability, statistics, and optimization. Additionally, continuous research and interest in applied mathematics created the need for the development of courses devoted entirely to another key concept, the functions of matrices.

There is a vast amount of references available focusing on the exponential function of a matrix, many of which are listed in the References section. While some of the references were used explicitly, all provided insight and assistance in the completion of this thesis. We begin now by defining key terms used throughout this thesis for clarity and cohesiveness.

1.1 Definitions

Before we begin our discussion of functions of matrices it is important to discuss some of the general terminology associated with matrices. We let $M_{m,n}$ (or M_n) denote the set of all $m \times n$ (or $n \times n$) complex matrices. We note that some authors use the notation $\mathbb{C}^{m \times n}$ (or $\mathbb{C}^{n \times n}$). Let $M_n(\mathbb{R})$ denote the set of all $n \times n$ real matrices. We denote the $n \times n$ *identity* matrix by I_n, or just by I.

Let $A \in M_n$. Then a nonzero vector $x \in \mathbb{C}^n$ is said to be an *eigenvector* of A corresponding to the *eigenvalue* λ if

$$Ax = \lambda x.$$

If the *characteristic polynomial of* A is defined by $p(\lambda) = \det(A - \lambda I)$, then the characteristic equation is

$$\det(A - \lambda I) = \det \begin{bmatrix} a_{11} - \lambda & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} - \lambda & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} - \lambda \end{bmatrix} = 0.$$

The set of all eigenvalues of A is called *the spectrum of* A and is denoted, $\sigma(A)$.

Let $p(t) = a_k t^k + a_{k-1} t^{k-1} + ... + a_1 t + a_0$. If $A \in M_n$, then
$p(A) = a_k A^k + a_{k-1} A^{k-1} + ... + a_1 A + a_0 I$.
Suppose $Ax = \lambda x, x \neq 0,\ x \in \mathbb{C}^n$. Then

$$\begin{aligned} p(A)x &= a_k A^k x + a_{k-1} A^{k-1} x + ... + a_1 Ax + a_0 Ix \\ &= a_k \lambda^k x + a_{k-1} \lambda^{k-1} x + ... + a_1 \lambda x + a_0 x \\ &= (a_k \lambda^k + a_{k-1} \lambda^{k-1} + ... + a_1 \lambda + a_0)x \\ &= p(\lambda)x. \end{aligned}$$

Therefore p(λ) is an *eigenvalue* of matrix $p(A)$ with corresponding *eigenvector* x.

The *minimal polynomial of* A, m(λ), is the unique monic polynomial of smallest degree such that $m(A) = 0$.

A matrix $D = [d_{ij}] \in M_n$ is called a *diagonal matrix* if $d_{ij} = 0$ whenever $i \neq j$.

Let $A, B \in M_n$. Then A is *similar* to B, denoted $A \sim B$, if there is a nonsingular matrix S such that $S^{-1}AS = B$. If $A \sim B$, then they have the same characteristic polynomial and therefore the same eigenvalues with the same multiplicities.

Let $A \in M_n$. Then A is *diagonalizable* if A is similar to a diagonal matrix.

Let $A \in M_n$. The *transpose of* $A = [a_{ij}]$ is a matrix $A^T \in M_n$ defined by $A^T = [a_{ji}]$.

Let $A \in M_n$. The *conjugate transpose of* A, denoted by A^*, is a matrix $A^* = \bar{A}^T = [\bar{a}_{ji}]$.

A matrix $A \in M_n(\mathbb{R})$ is *orthogonal* if $AA^T = I$.

A matrix $U \in M_n$ is *unitary* if $U^*U = I$.

A matrix $A \in M_n$ is *unitarily equivalent* or *unitarily similar* to $B \in M_n$ if there is an unitary matrix $U \in M_n$ such that $U^*AU = B$. If U may be taken to be real (and therefore real orthogonal), then A is (real) *orthogonally equivalent* to B.

If a matrix $A \in M_n$ is unitarily equivalent to a diagonal matrix, A is *unitarily diagonalizable*.

A matrix $A \in M_n(\mathbb{R})$ is *real symmetric* if $A^T = A$.

A matrix $A \in M_n(\mathbb{R})$ is *real skew-symmetric* if $A^T = -A$, so $a_{ij} = -a_{ji}$.

Let $A \in M_n$. A is *Hermitian* if $A^* = A$. If $A \in M_n$ is Hermitian, then the following statements hold:

(a) All eigenvalues of A are real; and

(b) A is unitarily diagonalizable.

Let $A \in M_n$. A matrix A is *skew-Hermitian* if $A^* = -A$.

Let $A \in M_n$. A is *upper triangular* if $a_{ij} = 0$ for $i > j$, i.e. all of the entries below the main diagonal are zero.

Let $A \in M_n$. A is *lower triangular* if $a_{ij} = 0$ for $i < j$, i.e. all of the entries above the main diagonal are zero.

A matrix N is *nilpotent* if $N^q = 0$ for some integer q.

1.2 Examples of General Matrix Functions

While the most common matrix function is the matrix inverse (usually mentioned with terms: invertible or nonsingular), other general matrix functions are the matrix square root, the exponential, the logarithm, and the trigonometric functions. The following are the definitions of the matrix functions mentioned above.

A matrix A is *invertible* or *nonsingular*, if there exists a unique *inverse* denoted by A^{-1}, where $A^{-1}A = I$ and $AA^{-1} = I$.

Let $A, B \in M_n$. Then B is a *square root* of A, if $B^2 = A$.

The *exponential of* $A \in M_n$, denoted e^A or $\exp(A)$, is defined by

$$e^A = I + A + \frac{A^2}{2!} + \cdots + \frac{A^k}{k!} + \cdots .$$

Let $A \in M_n$. Any X such that $e^X = A$ is a *logarithm* of A.

The *sine* and *cosine* of $A \in M_n$ are defined by

$$\cos(A) = I - \frac{A^2}{2!} + \cdots + \frac{(-1)^k}{(2k)!}A^{2k} + \cdots ,$$

$$\sin(A) = A - \frac{A^3}{3!} + \cdots + \frac{(-1)^k}{(2k+1)!}A^{2k+1} + \cdots .$$

2 FUNCTIONS OF MATRICES

Matrix functions are used throughout different areas of linear algebra and arise in numerous applications in science and engineering. Inverse, square root, logarithmic, sine and cosine functions can be defined using power series or as the solution of nonlinear systems. A general theory exists from which a number of properties possessed by all matrix functions can be deduced and which suggests different computational methods.

We provide a detailed overview of the basic ideas of functions of matrices to aid the reader in the understanding of the "connectivity" of the fundamental principles (many of which are defined in the introduction) of matrix theory. It was shown that if $Ax = \lambda x$ and $p(t)$ is a polynomial, then $p(A)x = p(\lambda)x$, so that if x is an eigenvector of A corresponding to λ, then x is an eigenvector of $p(A)$ corresponding to the eigenvalue $p(\lambda)$. We will shortly obtain an even stronger result.

Perhaps the most fundamentally useful fact of elementary matrix theory is that any matrix $A \in M_n$ is unitarily equivalent to an upper triangular (also to a lower triangular) matrix T. Representing the simplest form achievable under unitary equivalence, we now recall one of the most useful theorems in all of matrix theory, Schur's Theorem.

Schur's Theorem: If $A \in M_n$, then A is unitarily triangularizable, that is, there exists a unitary matrix U and an upper-triangular matrix T such that $U^*AU = T$.

Through the use of Schur's Theorem, one can prove that if $A \in M_n$ with $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $p(t)$ is a polynomial, then

$$\sigma(p(A)) = \{p(\lambda_1), \ldots, p(\lambda_n)\}.$$

The proof goes as follows: $U^*p(A)U = p(U^*AU) = p(T)$, which is upper-triangular with $p(\lambda_1), \ldots, p(\lambda_n)$ on the diagonal. The result follows from the similarity of $p(A)$ and $p(T)$.

We now shift our focus from polynomials to general functions.

Let $A \in M_n$ and suppose that λ_1, λ_2,..., λ_s are the distinct eigenvalues of A, so that

$$m(t) = (t-\lambda_1)^{m_1}(t-\lambda_2)^{m_2}\cdots(t-\lambda_s)^{m_s}$$

is the minimal polynomial of A with degree $m = m_1 + m_2 + \ldots + m_s$. Then m_k is the *index* of the eigenvalue λ_k, i.e. it is the size of the largest Jordan block associated with λ_k and is equal to the maximal degree of the elementary divisors associated with λ_k $(1 \leq k \leq s)$.

Now, a function $f(t)$ is *defined on the spectrum of* A if the numbers

$$f(\lambda_k), f'(\lambda_k), \ldots, f^{(m_k-1)}(\lambda_k), \quad k = 1, 2, ..., s,$$

are defined (exist). These numbers are called the *values of* $f(t)$ *on the spectrum of* A, where if $m_k = 1$, $f^{(m_k-1)}$ is $f^{(0)}$ or simply f.

Many of the succeeding results can be found in [12], but we will provide more details here.

Proposition 2.1: Every polynomial is defined on the spectrum of any matrix in M_n. For the polynomial $m(t)$, the values of

$$m(\lambda_k), m'(\lambda_k), ..., m^{(m_{k-1})}(\lambda_k), \quad k = 1, 2, \ldots, s,$$

are all zero.

Proof: The first statement is clear. Next, each $m(\lambda_k) = 0$. So,

$$m'(t) = (t-\lambda_1)^{m_1}\frac{d}{dt}[(t-\lambda_2)^{m_2}\cdots(t-\lambda_s)^{m_s}]+[(t-\lambda_2)^{m_2}\cdots(t-\lambda_s)^{m_s}]\cdot m_1(t-\lambda_1)^{m_1-1}.$$

Therefore,

$$m'(\lambda_1) = 0\cdot\frac{d}{dt}[(t-\lambda_2)^{m_2}\cdots(t-\lambda_s)^{m_s}]+[(t-\lambda_2)^{m_2}\cdots(t-\lambda_s)^{m_s}]\cdot 0 = 0, \text{if } m_1 > 1.$$

Similarly, for the other λ_k and the higher order derivatives. ■

Proposition 2.2: For the two polynomials $p_1(t)$ and $p_2(t)$, $p_1(A) = p_2(A)$ if and only if $p_1(t)$ and $p_2(t)$ have the same values on the spectrum of A.

Proof: $\Rightarrow$ Suppose $p_1(A) = p_2(A)$. Let $p_0(t) = p_1(t) - p_2(t)$. Then, $p_0(A) = 0$. So, $m(t)$ is a factor of $p_0(t)$, i.e. $p_0(t) = q(t)m(t)$ for some polynomial $q(t)$. Now, each term of $p_0^{(j)}(t)$ is a product, which involves one of the terms:

$$m(t), m'(t), ..., m^{(j)}(t).$$

Hence, by Proposition 2.1,

$$p_1^{(j)}(\lambda_k) - p_2^{(j)}(\lambda_k) = p_0^{(j)}(\lambda_k) = 0,$$

for $j = 0, 1, ..., m_k - 1,$ and $1 \leq k \leq s$. So, $p_1^{(j)}(\lambda_k) = p_2^{(j)}(\lambda_k)$ for the values of j and k.

$\Leftarrow$ We assume that $p_1(t)$ and $p_2(t)$ have the same values on the spectrum of A. Let $p_0(t) = p_1(t) - p_2(t)$, then

$$p_0^{(j)}(\lambda_k) = 0 \quad \text{for} \quad j = 0, 1, 2, ..., m_k - 1.$$

So, λ_k is a zero of $p_0(t)$ with multiplicity of at least m_k, i.e. $(t - \lambda_k)^{m_k}$ is a factor of $p_0(t)$. Hence, $m(t)$ is a factor of $p_0(t)$, where $p_0(t) = q(t)m(t)$ and therefore, $p_0(A) = 0$. Thus, $p_1(A) = p_2(A)$. ■

Proposition 2.3 (Interpolatory Polynomial): Given distinct numbers $\lambda_1, \lambda_2, \ldots, \lambda_s$, positive integers $m_1, m_2, \ldots, m_s$ with $m = \sum_{k=1}^{s} m_k$, and a set of numbers

$$f_{k,0}, \quad f_{k,1}, \ldots, \quad f_{k,m_k-1}, \; k = 1, 2, \ldots, s,$$

there exists a polynomial $p(t)$ of degree less than m such that

$$p(\lambda_k) = f_{k,0}, \quad p^{(1)}(\lambda_k) = f_{k,1}, \; \ldots, \quad p^{(m_k-1)}(\lambda_k) = f_{k,m_k-1} \quad \text{for} \quad k = 1, 2, \ldots, s. \tag{1}$$

Proof: It is easily seen that the polynomial $p_k(t) = \alpha_k(t)\psi_k(t)$ (note: if $s = 1$, then by definition $\psi_1(t) \equiv 1$), where $1 \le k \le s$ and

$$\alpha_k(t) = \alpha_{k,0} + \alpha_{k,1}(t-\lambda_k) + \cdots + \alpha_{k,m_k-1}(t-\lambda_k)^{m_k-1},$$

$$\psi_k(t) = \prod_{j=1, j\neq k}^{s} (t-\lambda_j)^{m_j},$$

has degree less than m and satisfies the conditions

$$p_k(\lambda_i) = p_k^{(1)}(\lambda_i) = \cdots = p_k^{(m_i-1)}(\lambda_i) = 0$$

for $i \neq k$ and arbitrary $\alpha_{k,0}, \alpha_{k,1}, \cdots, \alpha_{k,m_k-1}$. Hence, the polynomial

$$p(t) = p_1(t) + p_2(t) + \cdots + p_s(t) \tag{2}$$

satisfies conditions (1) if and only if

$$p_k(\lambda_k) = f_{k,0}, \quad p_k^{(1)}(\lambda_k) = f_{k,1}, \ \ldots, \quad p_k^{(m_k-1)}(\lambda_k) = f_{k,m_k-1} \text{ for each } 1 \le k \le s. \tag{3}$$

By differentiation,

$$p_k^{(j)}(\lambda_k) = \sum_{i=0}^{j} \binom{j}{i} \alpha_k^{(i)}(\lambda_k)\psi_k^{(j-i)}(\lambda_k)$$

for $1 \le k \le s,\ 0 \le j \le m_k - 1$. Using Eqs.(3) and recalling the definition of $\alpha_k(\lambda)$, we have for $k = 1, 2, \ldots, s,\ j = 0, 1, \ldots, m_k - 1$,

$$f_{k,j} = \sum_{i=0}^{j} \binom{j}{i} i!\alpha_{k,i}\psi_k^{(j-i)}(\lambda_k). \tag{4}$$

Since $\psi_k(\lambda_k) \neq 0$ for each fixed k, Eqs. (4) can now be solved successively (beginning with $j = 0$) to find the coefficients $\alpha_{k,0}, \ldots, \alpha_{k,m_k-1}$ for which (3) holds. Thus, a polynomial $p(t)$ of the form given in (2) satisfies the required conditions. ■

The interpolatory polynomial referred to in Proposition 2.3 is known as the *Hermite interpolating polynomial*. It is in fact unique, but the proof of the uniqueness is omitted, since it is quite cumbersome. If $f(t)$ is defined on the spectrum of A, we define $f(A)$ to be $p(A)$, where $p(t)$ is the interpolating polynomial for $f(t)$ on the spectrum of A.

Theorem 2.4: If $A \in M_n$ is a block-diagonal matrix,

$$A = \operatorname{diag}[A_1, A_2, ..., A_k],$$

and the function $f(t)$ is defined on the spectrum of A, then

$$f(A) = \operatorname{diag}[f(A_1), f(A_2), ..., f(A_k)]. \tag{5}$$

Proof: It is clear that for any polynomial $q(t)$,

$$q(A) = \operatorname{diag}[q(A_1), q(A_2), ..., q(A_k)].$$

Hence, if $p(t)$ is the interpolatory polynomial for $f(t)$ on the spectrum of A, we have

$$f(A) = p(A) = \operatorname{diag}[p(A_1), p(A_2), ..., p(A_k)].$$

Since the spectrum of A_j $(1 \leq j \leq k)$ is obviously a subset of the spectrum of A, the function $f(t)$ is defined on the spectrum of A_j for each $j = 1, 2, ..., k$. (Note also that the index of an eigenvalue of A_j cannot exceed the index of the same eigenvalue of A.) Furthermore, since $f(t)$ and $p(t)$ assume the same values on the spectrum of A, they must also have the same values on the spectrum of A_j $(j = 1, 2, ..., k)$. Hence,

$$f(A_j) = p(A_j)$$

and we obtain Eq. (5). ∎

Theorem 2.5: If $A, B, S \in M_n$, where $B = SAS^{-1}$, and $f(t)$ is defined on the spectrum of A, then

$$f(B) = Sf(A)S^{-1}. \tag{6}$$

Proof: Since A and B are similar, they have the same minimal polynomial. Thus, if $p(t)$ is the interpolatory polynomial for $f(t)$ on the spectrum of A, then it is also the interpolatory polynomial for $f(t)$ on the spectrum of B. Thus, we have

$$f(A) = p(A),$$

$$f(B) = p(B),$$

$$p(B) = Sp(A)S^{-1},$$

so the relation (6) follows. ■

Theorem 2.6: Let $A \in M_n$ and let $J = \text{diag}[J_j]_{j=1}^{k}$ be the Jordan canonical form of A, where $A = SJS^{-1}$ and J_j is the j^{th} Jordan block of J. Then

$$f(A) = S\ \text{diag}[f(J_1), f(J_2), ..., f(J_k)]S^{-1}. \tag{7}$$

The last step in computing $f(A)$ by use of the Jordan form of A consists of the following formula.

Theorem 2.7: Let J_0 be a Jordan block of size l associated with λ_0:

$$J_0 = \begin{bmatrix} \lambda_0 & 1 & & \\ & \lambda_0 & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_0 \end{bmatrix}.$$

If $f(t)$ is an $(l-1)$-times differentiable function in a neighborhood of λ_0, then

$$f(J_0) = \begin{bmatrix} f(\lambda_0) & \frac{1}{1!}f'(\lambda_0) & \dots & \frac{1}{(l-1)!}f^{(l-1)}(\lambda_0) \\ 0 & f(\lambda_0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{1}{1!}f'(\lambda_0) \\ 0 & \dots & 0 & f(\lambda_0) \end{bmatrix}. \tag{8}$$

Proof: The minimal polynomial of J_0 is $(t-\lambda_0)^l$ and the values of $f(t)$ on the spectrum of J_0 are therefore $f(\lambda_0)$, $f'(\lambda_0)$,...,$f^{(l-1)}(\lambda_0)$. The interpolatory polynomial $p(t)$, defined by the values of $f(t)$ on the spectrum $\{\lambda_0\}$ of J_0, is found by putting $s = 1, m_k = l, \lambda_1 = \lambda_0$, and $\psi_1(t) \equiv 1$, in Eqs.(2) thru (4). One obtains

$$p(t) = \sum_{i=0}^{l-1} \frac{1}{i!} f^{(i)}(\lambda_0)(t - \lambda_0)^i.$$

The fact that the polynomial $p(t)$ solves the interpolation problem $p^{(j)}(\lambda_0) = f^{(j)}(\lambda_0), 1 \leq j \leq l - 1$, can also be easily checked by a straightforward calculation.

We then have $f(J_0) = p(J_0)$ and hence

$$f(J_0) = \sum_{i=0}^{l-1} \frac{1}{i!} f^{(i)}(\lambda_0)(J_0 - \lambda_0 I)^i.$$

Computing the powers of $J_0 - \lambda_0 I$, we obtain

$$(J_0 - \lambda_0 I)^i = \begin{bmatrix} 0 & 1 & & \\ & 0 & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{bmatrix}^i = \begin{bmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ & & & & & \ddots & \vdots \\ & & & & \ddots & & 0 \\ & & & & \ddots & & 1 \\ & & & & & & 0 \\ & & & & & & \vdots \\ & & & & & & 0 \end{bmatrix}$$

with 1's in the i-th super-diagonal positions, and zeros elsewhere, and Eq.(8) follows.

■

Thus, given a Jordan decomposition of the matrix A, the matrix $f(A)$ is easily found by combining Theorems 2.6 and 2.7.

From Theorems 2.6 and 2.7, we have the following results.

Theorem 2.8: Using the notation of Theorem 2.6,

$$f(A) = S \text{ diag}[f(J_1), f(J_2), ..., f(J_k)]S^{-1},$$

where $f(J_i)$ $(i = 1, 2, ..., k)$ are upper triangular matrices of the form given in Eq.(8).

Theorem 2.9: If $\lambda_1, \lambda_2, ..., \lambda_n$ are the eigenvalues of the matrix $A \in M_n$ and $f(t)$ is defined on the spectrum of A, then the eigenvalues of $f(A)$ are $f(\lambda_1), f(\lambda_2), ..., f(\lambda_n)$.

This follows from the fact that the eigenvalues of an upper triangular matrix are its diagonal entries.

3 EXPONENTIAL FUNCTION

The exponential function of matrices is a very important subclass of functions of matrices that has been studied extensively in the last 50 years, see [2, 4, 15]. In mathematics, the matrix exponential is a function on square matrices analogous to the ordinary exponential function. Let $A \in M_n$. The exponential of A, denoted by e^A or $\exp(A)$, is the $n \times n$ matrix given by the power series

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!}.$$

The above series always converges, so the exponential of A is well-defined. Note that if A is a 1×1 matrix, the matrix exponential of A corresponds with the ordinary exponential of A thought of as a number.

3.1 Properties

To make full use of the exponential function we must consider its properties as given in [7]. Let $A, B \in M_n$ and let t and s be arbitrary complex numbers. We denote the $n \times n$ *zero matrix* by $\mathbf{0}$. The matrix exponential satisfies the following properties:

Property 3.1.1: $e^{\mathbf{0}} = I$.

Property 3.1.2: If A is invertible, then $e^{ABA^{-1}} = Ae^BA^{-1}$.

Property 3.1.3: $\det(e^A) = e^{\mathrm{tr}(A)}$.

Property 3.1.4: $e^{(A^T)} = (e^A)^T$. It follows that if A is symmetric, then e^A is also symmetric, and that if A is skew-symmetric, then e^A is orthogonal.

Property 3.1.5: $e^{(A^*)} = (e^A)^*$. It follows that if A is Hermitian, then e^A is also Hermitian, and that if A is skew-Hermitian, then e^A is unitary.

Property 3.1.6: $(e^{At})' = Ae^{At}$.

Proof: Let $\mathbf{x}_0$ denote a column of the identity matrix. Define $\mathbf{x}(t) = e^{At}\mathbf{x}_0$. Then

$$\begin{aligned}(e^{At})'\mathbf{x}_0 &= \mathbf{x}'(t) \\ &= A\mathbf{x}(t) \\ &= Ae^{At}\mathbf{x}_0.\end{aligned}$$

Because this identity holds for all columns of the identity matrix, then $(e^{At})'$ and Ae^{At} have identical columns, hence we have proved the identity $(e^{At})' = Ae^{At}$. ■

A real valued function *f*, defined on a subset D of the real numbers

$$f : D \subseteq \mathbb{R} \to \mathbb{R}$$

is called *Lipschitz continuous* if there exists a constant $K \geq 0$ such that for all x_1, x_2 in D

$$| f(x_1) - f(x_2) | \leq K | x_1 - x_2 | .$$

Picard – Lindelöf Theorem: An initial value problem

$$y'(t) = f(t, y(t)), \quad y(t_o) = y_0,$$

has exactly one solution if f is Lipschitz continuous in y, continuous in t as long as $y(t)$ stays bounded.

Property 3.1.7: If AB = BA, then $e^{At}B = Be^{At}$.

Proof: Define $\mathbf{w}_1(t) = e^{At}B\mathbf{w}_0$ and $\mathbf{w}_2(t) = Be^{At}\mathbf{w}_0$. Calculate $\mathbf{w}_1'(t) = A\mathbf{w}_1(t)$ and $\mathbf{w}_2'(t) = BAe^{At}\mathbf{w}_0 = ABe^{At}\mathbf{w}_0 = A\mathbf{w}_2(t)$, due to $BA = AB$. Because $\mathbf{w}_1(0) = \mathbf{w}_2(0) = \mathbf{w}_0$, then the uniqueness assertion of the Picard-Lindelöf theorem implies that $\mathbf{w}_1(t) = \mathbf{w}_2(t)$. Because $\mathbf{w}_0$ is any vector, then $e^{At}B = Be^{At}$. ■

Property 3.1.8: If AB = BA, $e^{At}e^{Bt} = e^{(A+B)t}$.

Proof: Let $\mathbf{x}_0$ be a column of the identity matrix. Define $\mathbf{x}(t) = e^{At}e^{Bt}\mathbf{x}_0$ and $\mathbf{y}(t) = e^{(A+B)t}\mathbf{x}_0$. We must show that $\mathbf{x}(t) = \mathbf{y}(t)$ for all t. Define $\mathbf{u}(t) = e^{Bt}\mathbf{x}_0$. We will apply the result $e^{At}B = Be^{At}$, valid for $BA = AB$, that we have shown in the proof of property 3.1.7. The details:

$$\begin{aligned}
\mathbf{x}'(t) &= (e^{At}\mathbf{u}(t))' \\
&= Ae^{At}\mathbf{u}(t) + e^{At}\mathbf{u}'(t) \\
&= Ae^{At}e^{Bt}\mathbf{x}_0 + e^{At}Be^{Bt}\mathbf{x}_0 \\
&= A\mathbf{x}(t) + e^{At}B\mathbf{u}(t) \\
&= A\mathbf{x}(t) + Be^{At}\mathbf{u}(t) \\
&= A\mathbf{x}(t) + Be^{At}e^{Bt}\mathbf{x}_0 \\
&= A\mathbf{x}(t) + B\mathbf{x}(t) \\
&= (A+B)\mathbf{x}(t).
\end{aligned}$$

We also know that $\mathbf{y}'(t) = (A+B)\mathbf{y}(t)$ and since $\mathbf{x}(0) = \mathbf{y}(0) = \mathbf{x}_0$, then the Picard-Lindelöf theorem implies that $\mathbf{x}(t) = \mathbf{y}(t)$ for all t. ■

Property 3.1.9: $e^{A}e^{-A} = I$.

Property 3.1.10: $e^{At}e^{As} = e^{A(t+s)}$.

Proof: Let t be a variable and consider s fixed. Define $\mathbf{x}(t) = e^{At}e^{As}\mathbf{x}_0$ and $\mathbf{y}(t) = e^{A(t+s)}\mathbf{x}_0$. Then $\mathbf{x}(0) = \mathbf{y}(0)$ and both satisfy the differential equation $\mathbf{u}'(t) = A\mathbf{u}(t)$ because $\mathbf{x}'(t) = Ae^{At}e^{As}\mathbf{x}_0 = A\mathbf{x}(t)$ and $\mathbf{y}'(t) = Ae^{A(t+s)}\mathbf{x}_0 = A\mathbf{y}(t)$. By the uniqueness in the Picard-Lindelöf theorem, $\mathbf{x}(t) = \mathbf{y}(t)$, which implies $e^{At}e^{As} = e^{A(t+s)}$. ■

3.2 Usefulness

The exponential of a matrix can be used in various fields. One of the reasons for the importance of the matrix exponential is that it can be used to solve systems of linear ordinary differential equations.

Let $A \in M_n$ and $y(t) = e^{At}\mathbf{y}_0$. Then

$$\frac{d}{dt}y(t) = Ay(t), \quad y(0) = \mathbf{y}_0,$$

is given by

$$y(t) = e^{At}\mathbf{y}_0.$$

Another use of the matrix exponential is that mathematical models of many physical, biological, and economic processes involve systems of linear ordinary differential equations with constant coefficient,

$$x'(t) = A\mathbf{x}(t).$$

Here $A \in M_n$ is a given fixed matrix. A solution vector $\mathbf{x}(t)$ is sought, which satisfies an initial condition

$$x(0) = \mathbf{x}_0.$$

In control theory, A is known as the state companion matrix and $\mathbf{x}(t)$ is the system response. In principal, the solution is given by $x(t) = e^{tA}\mathbf{x}_0$, where e^{tA} can be formally defined by the convergent power series

$$e^{tA} = \sum_{k=0}^{\infty} \frac{t^k A^k}{k!}.$$

The matrix exponential can also be used to solve the inhomogeneous equation

$$\frac{d}{dt}y(t) = Ay(t) + z(t), \quad y(0) = y_0.$$

4 COMPUTATIONS

There are many methods used to compute the exponential of a matrix. Approximation theory, differential equations, the matrix eigenvalues, and the matrix characteristic polynomial are some of the various methods used. In practice, consideration of computational stability and efficiency indicates that some of the methods are preferable to others, but that none are completely satisfactory.

4.1 Effectiveness

The following is adapted from [16]. In assessing the effectiveness of various algorithms the following criteria are normally used: generality, reliability, stability, accuracy, and efficiency. Other characteristics such as storage requirements, ease of use, and simplicity may also be taken into account. Now we will give a brief description of some of the more commonly used criteria.

An algorithm is *general* if the method is applicable to wide classes of matrices.

An algorithm is *reliable* if it gives some warning whenever it introduces excessive errors.

An algorithm is *stable* if it does not introduce any more sensitivity to a disturbance of arrangement than is inherent in the underlying problem.

An algorithm is *accurate* if the error introduced by truncating infinite series or terminating iterations is minimal.

An algorithm is *efficient* if the amount of computer time required to solve a particular problem is reasonable.

An algorithm is considered completely satisfactory if it could be used as the basis for a general purpose subroutine. This would mean that an algorithm meets the standards of quality now available for linear algebraic equations, matrix eigenvalues, and the initial value problems for nonlinear ordinary differential equations.

4.2 Special Cases of Computing the Matrix Exponential

In this section of the thesis we will outline various simplistic methods for finding the exponential of a matrix. The methods examined are given by the type of matrix. Here we examine diagonal matrices, nilpotent matrices, matrices that can be written as a sum of diagonalizable and nilpotent matrices that commute, and 2×2 matrices.

4.2.1 Diagonalizable case

If a matrix is diagonal:

$$A = \begin{bmatrix} a_1 & 0 & \dots & 0 \\ 0 & a_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_n \end{bmatrix},$$

then its exponential can be obtained by just exponentiating every entry on the main diagonal:

$$e^A = \begin{bmatrix} e^{a_1} & 0 & \dots & 0 \\ 0 & e^{a_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{a_n} \end{bmatrix}.$$

This also allows one to exponentiate diagonalizable matrices. If $A = UDU^{-1}$ and D is diagonal, then $e^A = Ue^DU^{-1}$.

4.2.2 Nilpotent Case

Let $N^q = 0$ for $q \in \mathbb{N}$. In this case, the matrix exponential e^N can be computed directly from the series expansion, as the series terminates after a finite number of terms:

$$e^N = I + N + \frac{1}{2}N^2 + \frac{1}{6}N^3 + \cdots + \frac{1}{(q-1)!}N^{q-1}.$$

4.2.3 Commutable Case

An arbitrary matrix X (over an algebraically closed field) can be expressed uniquely as sum

$$X = A + N,$$

where

- A is diagonalizable
- N is nilpotent
- A commutes with N (i.e. AN = NA)

This means we can compute the exponential of X by reduction to the previous two cases:

$$e^X = e^{A+N} = e^A e^N.$$

Note that we need the commutability of A and N for the last step to work. This may seem simple, but it is not always a possibility to determine the appropriate A and N due to stability. The problem of computing A and N is essentially equivalent to determining the Jordan structure of X, which is known to be a highly unstable computation. Here we will examine an example using the Jordan Canonical form as our method for writing our matrix as a sum of diagonalizable and nilpotent matrices that commute.

Suppose that we want to compute the exponential of

$$B = \begin{bmatrix} 21 & 17 & 6 \\ -5 & -1 & -6 \\ 4 & 4 & 16 \end{bmatrix}.$$

Its Jordan form is $J = PBP^{-1}$

$$= \begin{bmatrix} 4 & 0 & 0 \\ 0 & 16 & 1 \\ 0 & 0 & 16 \end{bmatrix},$$

where the transition matrix is given by

$$P = \begin{bmatrix} -\frac{1}{4} & 2 & \frac{5}{4} \\ \frac{1}{4} & -2 & -\frac{1}{4} \\ 0 & 4 & 0 \end{bmatrix}.$$

Let us first calculate $\exp(J)$. We have

$$J = J_1(4) \oplus J_2(16).$$

Now we can use the method of finding the exponential of the sum. The exponential of a 1×1 matrix is just the exponential of the one entry of the matrix, so $\exp(J_1(4))$ $= e^4$. The exponential of $J_2(16)$ can be calculated by the formula $e^{(\lambda I + N)} = e^\lambda e^N$ mentioned above, where

$$\lambda I = \begin{bmatrix} 16 & 0 \\ 0 & 16 \end{bmatrix} \text{ and } N = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

This yields

$$\begin{aligned} \exp\left(\begin{bmatrix} 16 & 1 \\ 0 & 16 \end{bmatrix}\right) &= e^{16} \exp\left(\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\right) \\ &= e^{16}\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + \frac{1}{2!}\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \cdots\right) \\ &= \begin{bmatrix} e^{16} & e^{16} \\ 0 & e^{16} \end{bmatrix}. \end{aligned}$$

Therefore, the exponential of the original matrix B is

$$\begin{aligned}\exp(B) &= P\exp(J)P^{-1}\\ &= P\begin{bmatrix} e^4 & 0 & 0\\ 0 & e^{16} & e^{16}\\ 0 & 0 & e^{16}\end{bmatrix}P^{-1}\\ &= \frac{1}{4}\begin{bmatrix} 13e^{16}-e^4 & 13e^{16}-5e^4 & 2e^{16}-2e^4\\ -9e^{16}+e^4 & -9e^{16}+5e^4 & -2e^{16}+2e^4\\ 16e^{16} & 16e^{16} & 4e^{16}\end{bmatrix}.\end{aligned}$$

Clearly, to calculate the Jordan form and to evaluate the exponential this way is very tedious for matrices of bigger sizes. Unfortunately, the Jordan block structure of a defective matrix are very difficult to determine numerically. Small changes in a defective matrix can radically alter its Jordan form. A single rounding error may cause some multiple eigenvalues to become distinct or vice versa altering the entire structure of J and P. Therefore there are limitations of Jordan decomposition in numerical analysis. Fortunately, the stable Schur decomposition can almost always be used in lieu of Jordan decomposition in practical applications.

4.2.4 2×2 Case

In the case of 2×2 real matrices we have a simplistic way of computing the matrix exponential. The eigenvalues of matrix A are the roots of the characteristic polynomial $\lambda^2 - \mathrm{tr}(A)\lambda + \det(A) = 0$. The discriminant D is computed by $\mathrm{tr}(A)^2 - 4\det(A)$. The discriminant will be used to differentiate between the three cases which are used to compute the matrix exponential of a 2×2 matrix.

Case 1: D > 0

The matrix A has real distinct eigenvalues λ_1, λ_2 with eigenvectors $\mathbf{v}_1$, $\mathbf{v}_2$;

$$e^{At} = \begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2\end{bmatrix}\begin{bmatrix} e^{\lambda_1 t} & 0\\ 0 & e^{\lambda_2 t}\end{bmatrix}\begin{bmatrix}\mathbf{v}_1 & \mathbf{v}_2\end{bmatrix}^{-1}.$$

Example of case 1.

$$A = \begin{bmatrix} 4 & -2 \\ 1 & 1 \end{bmatrix}$$

Here det(A) = 6 and tr(A) = 5, which means D = 1. The characteristic equation is

$$\lambda^2 - 5\lambda + 6 = 0.$$

The eigenvalues are 2 and 3, and the eigenvectors are $\begin{bmatrix} 1 & 1 \end{bmatrix}^T$ and $\begin{bmatrix} 2 & 1 \end{bmatrix}^T$, respectively. Therefore

$$\begin{aligned} e^A &= \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} e^2 & 0 \\ 0 & e^3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} -e^2 + 2e^3 & 2e^2 - 2e^3 \\ -e^2 + e^3 & 2e^2 - e^3 \end{bmatrix} \\ &= \begin{bmatrix} 32.7820 & -25.3930 \\ 12.6965 & -5.3074 \end{bmatrix}. \end{aligned}$$

Case 2: D = 0

The matrix A has a real double eigenvalue λ.

If $A = \lambda I$, then:

$$e^{At} = e^{\lambda t} I,$$

otherwise:

$$e^{At} = \begin{bmatrix} \mathbf{v} & \mathbf{w} \end{bmatrix} e^{\lambda t} \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{v} & \mathbf{w} \end{bmatrix}^{-1},$$

where $\mathbf{v}$ is an eigenvector of A and $\mathbf{w}$ satisfies $(A - \lambda I)\mathbf{w} = \mathbf{v}$.

Example of Case 2.

$$A = \begin{bmatrix} 6 & -1 \\ 4 & 2 \end{bmatrix}$$

Here det(A) = 16 and tr(A) = 8, therefore D = 0. The characteristic equation is

$$\lambda^2 - 8\lambda + 16 = 0,$$

thus $\lambda = 4$. The eigenvector associated with the eigenvalue 4 is $\mathbf{v} = \begin{bmatrix} 1 & 2 \end{bmatrix}^T$. Solving

$$\left(\begin{bmatrix} 6 & -1 \\ 4 & 2 \end{bmatrix} - \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \right) \mathbf{w} = \begin{bmatrix} 1 \\ 2 \end{bmatrix},$$

we obtain $\mathbf{w} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. Using the method for 2×2 matrices with a double eigenvalue, we have found,

$$\begin{aligned} e^A &= \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix} e^4 \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}^{-1} \\ &= e^4 \begin{bmatrix} 3 & -1 \\ 4 & -1 \end{bmatrix} \\ &= \begin{bmatrix} 3e^4 & -e^4 \\ 4e^4 & -e^4 \end{bmatrix} \\ &= \begin{bmatrix} 163.7945 & -54.5982 \\ 218.3926 & -54.5982 \end{bmatrix}. \end{aligned}$$

Case 3: D $<$ 0

The matrix A has complex conjugate eigenvalues $\lambda, \bar{\lambda}$ with eigenvectors $\mathbf{u}, \bar{\mathbf{u}}$;

$$e^{At} = \begin{bmatrix} \mathbf{u} & \bar{\mathbf{u}} \end{bmatrix} \begin{bmatrix} e^{\lambda t} & 0 \\ 0 & e^{\bar{\lambda} t} \end{bmatrix} \begin{bmatrix} \mathbf{u} & \bar{\mathbf{u}} \end{bmatrix}^{-1},$$

or writing $\lambda = \sigma + i\omega$, $\mathbf{u} = \mathbf{v} + i\mathbf{w}$,

$$e^{At} = \begin{bmatrix} \mathbf{v} & \mathbf{w} \end{bmatrix} e^{\sigma t} \begin{bmatrix} \cos \omega t & -\sin \omega t \\ \sin \omega t & \cos \omega t \end{bmatrix} \begin{bmatrix} \mathbf{v} & \mathbf{w} \end{bmatrix}^{-1}$$

Example of Case 3.

$$A = \begin{bmatrix} 3 & -2 \\ 1 & 1 \end{bmatrix}$$

Since det(A) = 5 and tr(A) = 4, D = -4. The characteristic equation is

$$\lambda^2 - 4\lambda + 5 = 0,$$

and $\lambda = 2 \pm i$. The eigenvector $\mathbf{u} = \begin{bmatrix} 2 & 1 - i \end{bmatrix}^T$. Therefore $\sigma = 2$, $\omega = 1$, $\mathbf{v} = \begin{bmatrix} 2 & 1 \end{bmatrix}^T$ and $\mathbf{w} = \begin{bmatrix} 0 & -1 \end{bmatrix}^T$. So

$$\begin{aligned}
e^A &= \begin{bmatrix} 2 & 0 \\ 1 & -1 \end{bmatrix} e^2 \begin{bmatrix} \cos 1 & -\sin 1 \\ \sin 1 & \cos 1 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 1 & -1 \end{bmatrix}^{-1} \\
&= e^2 \begin{bmatrix} \cos 1 - \sin 1 & -2\sin 1 \\ -\sin 1 & \sin 1 + \cos 1 \end{bmatrix} \\
&= \begin{bmatrix} -2.2254 & -12.4354 \\ 6.2177 & 10.21 \end{bmatrix}.
\end{aligned}$$

4.3 Computational Methods

For the last 50 years there have been various methods for computing e^{At}. Results from analysis, approximation theory and matrix theory have been used to obtain some of these methods. The paper *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*, by C. Moler and C. Van Loan [16] details various methods for computing the matrix exponential some of which will be highlighted here. We will focus on the Series method and Schur Decomposition. By the standards given in the effectiveness section, none of the methods are completely satisfactory.

4.3.1 Scaling and Squaring

A fundamental property unique to any exponential function is $(e^{a/b})^b = e^a$, where a and b are scalars. These property can be applied to matrices such that

$$e^A = (e^{A/m})^m, \text{where } A \in M_n \text{ and } m \text{ is a positive integer.}$$

This method will help to control some of the round off error and time or number of terms it would take to find a Taylor approximation. These factors are affected as the spread of the eigenvalues of A increases. The idea is to choose m to be a power of two for which $e^{A/m}$ can be computed, and then to form the matrix $(e^{A/m})^m$ by repeated squarings. One commonly used criteria for choosing m is to make it the smallest power of two for which $\|A\|/m \leq 1$. With this restriction, $e^{A/m}$ can be computed by Taylor approximation. The Taylor approximation alone is not a satisfactory method. Round off error as well as sign issues when rounding make this method unreliable. However when used in conjuction with the scaling and squaring method, the resulting algorithm is one of the most effective known.

Below is a script file (as it should be typed for use in MATLAB) to find the exponential of a matrix using scaling and squaring: n = input ('Enter the size of a matrix: ')

```
T = input (' Enter the matrix n × n: ')
n0 = norm (T);
for k = 1:50
k0 = k;
if 2^ k > 2*n0
break
end
end
A = T/2^k0;
for i = 1:50
i0 = i;
nk = norm((A^i)/prod(1:i));
if nk<10^(-16)
break
end
end
M0 = eye(n,n);
for i = 1:i0
M0 = M0 + (A^i)/prod(1:i);
end
M = M0^ (2^k0);
E0 = expm(T);
norm(M-E0)/norm(E0)
k0
i0
```

Explanation of the program:

n = input ('Enter the size of a matrix: '): Indicate the size of the matrix

T = input (' Enter the matrix $n \times n$: ') : Enter the entries of the matrix enclosed in brackets. Separate each entry with a comma and each row with a semicolon.

n0 = norm (T); : Compute the norm of the inputed matrix

for k = 1:50

k0 = k;

if 2^ k > 2 * n0

break

end

end :

A = T/2^k0;

The loop above identifies the value of k_0 such that $2^{k_0} > 2 * \text{norm(T)}$. Then $\|A\| = \|T/2^{k_0}\| < \frac{1}{2}$, and this improves convergence of the series as compared with the suggestion of $\|A\| < 1$.

for i = 1:50

i0 = i;

nk = norm((A^ i)/prod(1:i));

if nk<10^(-16)

break

end

end:

In the loop above we find out how many terms should be included into partial sum of the exponential series. We stop when$\|\frac{A^i}{i!}\| < 10^{-16}$.

M0 = eye(n,n); eye(n,n) is the identity matrix of size $n \times n$.

for i = 1:i0

M0 = M0 + (A^i)/prod(1:i);

end :

The loop above computes the partial sum of the exponential series for e^A. The matrix M0 is the approximation for e^A.

M = M0^(2^k0); The matrix M is the approximation for $(e^A)^{2^{k_0}} = e^T$.

E0 = expm(T); This is the computation of the matrix exponential using the embedded MATLAB program.

norm(M-E0)/norm(E0):

Here we find the norm difference between the matrix exponential that we have computed using the scaling and squaring method previously described and the matrix exponential computed using the embedded MATLAB program, and then we compute a relative error.

k0 Here we are given the number of squaring.

i0 Here we are given the number of terms in the series.

Example of implementing the scaling and squaring MATLAB program.

$\gg$ ExpScaling (Name given to the program)

Enter the size of a matrix: 4

Output: n=4

Enter the matrix $n \times n$: rand(4,4) : We entered a random 4×4 matrix.

Output: T =

$$\begin{bmatrix} .3200 & .7446 & .6833 & .1338 \\ .9601 & .2679 & .2126 & .2071 \\ .7266 & .4399 & .8392 & .6072 \\ .4120 & .9334 & .6288 & .6299 \end{bmatrix}$$

Output: M =

$$\begin{bmatrix} 2.7527 & 1.8971 & 1.9496 & .9366 \\ 2.0283 & 2.3755 & 1.3333 & .8127 \\ 2.6324 & 2.2822 & 3.7697 & 1.8405 \\ 2.3538 & 2.6498 & 2.3682 & 2.7839 \end{bmatrix}$$

Output: E0 =

$$\begin{bmatrix} 2.7527 & 1.8971 & 1.9496 & .9366 \\ 2.0283 & 2.3755 & 1.3333 & .8127 \\ 2.6324 & 2.2822 & 3.7697 & 1.8405 \\ 2.3538 & 2.6498 & 2.3682 & 2.7839 \end{bmatrix}$$

Output: norm(M-E0)/norm(E0) = 1.1166e-015

Output: k0 = 3

Output: i0 = 13

According to the results given, the exponential computed using the scaling and squaring method of our program and the exponential computed using the MATLAB embedded program, the relative norm of the difference is of the order 10^{-15}.

4.3.2 Schur Method

The Schur decomposition

$$T = q\ t\ q^T$$

for real matrix T with real orthogonal q and real upper triangular t exists if T has real eigenvalues. If T has complex eigenvalues, then it is necessary to allow 2×2 blocks on the diagonal of t or to make q and t complex (and replace q^T with q^*). The Schur decomposition can be computed reliably and quite efficiently. Once the Schur decomposition is available,

$$e^T = q\ e^t\ q^*.$$

The only problematic part is the computation of e^t, where t is a triangular or quasitriangular matrix. Note that the eigenvectors of t are not required. In order to force the Schur Decomposition program to make complex output, we add the matrix $\epsilon i I$, where ϵ is a small real number and I is the identity matrix, to the real matrix T. In this case we will set an upper triangular matrix t and unitary matrix q. If t is upper triangular with diagonal elements $\lambda_1, \ldots, \lambda_n$, then it is clear that e^t is upper triangular with diagonal elements $e^{\lambda_1}, \ldots, e^{\lambda_n}$.

Below is a script file (as it should be typed for use in MATLAB) to find the exponential of a matrix using Schur Decomposition for matrices with distinct eigenvalues:

```
n = input( 'Enter the size of a matrix: ')
T = input ('Enter the matrix n × n:')
E0 = expm(T);
n0 = norm(T);
T0 = T + 0.0000001i * n0 * eye(n, n);
E1 = exp(-0.0000001i * n0) * eye(n, n);
[q,t] = schur(T0);
td = diag(t);
b = diag(exp(td));
for r = 1:n-1
for i = 1:n-r
j = i+r;
s = t(i,j) * (b(j,j)-b(i,i));
for k = i+1:j-1
s = s+t(i,k)*b(k,j)-b(i,k)*t(k,j);
end
b(i,j) = s/(t(j,j)-t(i,i));
end
end
```

$c = q * b * q' * E1;$

norm(c-E0)/norm(E0)

Explanation of the program:

n = input ('Enter the size of a matrix: '): Indicate the size of the matrix

T = input (' Enter the matrix $n \times n$: ') : Enter the entries of the matrix enclosed in brackets. Separate each entry with a comma and each row with a semicolon.

E0 = expm(T); Compute the exponential of T using the embedded MATLAB program.

n0 = norm(T); Compute the norm of the inputed matrix.

$T0 = T + 0.0000001i * n0 * \text{eye}(n, n);$

eye(n,n) is the $n \times n$ identity matrix. Here we give a signal to the embedded program of Schur decomposition that we are interested in complex, not real matrices, because we want to receive an upper triangular matrix as a result of Schur decomposition.

$E1 = \exp(-0.0000001i * n0) * \text{eye}(n, n);$

[q,t] = schur(T0);

q is a unitary matrix and t is an upper triangular matrix. This part of the program computes the Schur Decomposition of the matrix $T0 = q * t * q'$.

td = diag(t); Extract diagonal elements of the triangular matrix t.

b = diag(exp(td)); Computes the diagonal elements of the matrix e^t.

for r = 1:n-1

for i = 1:n-r

j = i+r;

s = t(i,j) * (b(j,j)-b(i,i));

for k = i+1:j-1

s = s+t(i,k)*b(k,j)-b(i,k)*t(k,j);

end

b(i,j) = s/(t(j,j)-t(i,i));

end

end:

The loop above computes the other elements of the matrix $b = e^t$. The computation progresses in the direction parallel to the main diagonal.

$c = q * b * q' * E1;$

Since $b = e^t$, we have $q\ b\ q' = e^{T0}$. We know that $T0 = T + \epsilon I$, where $\epsilon = 0.0000001i * n0$ and I = eye(n,n) is identity matrix. Therefore $e^{T0} = e^{T+\epsilon I} = e^T * e^{\epsilon I}$, because $(\epsilon I) * T = T * (\epsilon I)$, $e^T = e^{T0} * e^{-\epsilon I} = e^{T0} * E1 = qbq'E1$, where $E1 = e^{-\epsilon I}$.

norm(c-E0)/norm(E0):

Calculates the relative error between our program's method of computing the matrix exponential and the MATLAB embedded method of computing the matrix exponential.

Example of computing exponential of a matrix Schur Decomposition with MATLAB.

$\gg$ ExpSchur (Name given to the program)

Enter the size of a matrix: 4

Output: n = 4

Enter the matrix $n \times n$: rand (4,4) (We entered a random 4×4 matrix.)

Output: T =

$$\begin{bmatrix} .3200 & .7446 & .6833 & .1338 \\ .9601 & .2679 & .2126 & .2071 \\ .7266 & .4399 & .8392 & .6072 \\ .4120 & .9334 & .6288 & .6299 \end{bmatrix}$$

Output: n0 =

2.2493 (norm of T)

If we make Schur Decomposition of this matrix, the result is not an upper triangular matrix, because it has complex eigenvalues.

Output: t =

$$\begin{bmatrix} 2.1461 & -0.2717 & 0.4884 & 0.3416 \\ 0 & -0.6776 & -0.2608 & -0.0098 \\ 0 & 0 & 0.2943 & -0.1671 \\ 0 & 0 & 0.2770 & 0.2943 \end{bmatrix}$$

eig(T)

Output: ans =

2.1461

-0.6776

0.2943+0.2152i

0.2943-0.2152i

This is the reason that we added the imaginary part, using

$$T0 = T + 0.0000001i * n0 * eye(n, n),$$

to ensure that we will have an upper triangular matrix.

Output: T0=

$$\begin{bmatrix} 0.3200 + 0.0000i & 0.7446 & 0.6833 & 0.1338 \\ 0.9601 & 0.2679 + 0.0000i & 0.2126 & 0.2071 \\ 0.7266 & 0.4399 & 0.8392 + 0000i & 0.6072 \\ 0.4120 & 0.9334 & 0.6288 & 0.6299 + 0000i \end{bmatrix}$$

schur(T0)

Output: ans =

$$\begin{bmatrix} 2.1461 + 0.0000i & 0.2347 + 0.1368i & 0.3458 + 0.2169i & 0.4210 - 0.1491i \\ 0 & -0.6776 + 0000i & 0.1283 - 0.0958i & 0.1891 - 0.0819i \\ 0 & 0 & 0.2943 - 0.2152i & 0.0335 - 0.1047i \\ 0 & 0 & 0 & 0.2943 + 0.2152i \end{bmatrix}$$

Output: c =

$$\begin{bmatrix} 2.7527 & 1.891 & 1.9496 & 0.9366 \\ 2.0283 & 2.3755 & 1.3333 & 0.8127 \\ 2.6324 & 2.2822 & 3.7697 & 1.8405 \\ 2.3538 & 2.6498 & 2.3682 & 2.7839 \end{bmatrix}$$

Output: E0 =

$$\begin{bmatrix} 2.7527 & 1.891 & 1.9496 & 0.9366 \\ 2.0283 & 2.3755 & 1.3333 & 0.8127 \\ 2.6324 & 2.2822 & 3.7697 & 1.8405 \\ 2.3538 & 2.6498 & 2.3682 & 2.7839 \end{bmatrix}$$

norm(c-E0)/norm(E0)

Output: ans =

7.1296e-015

So our approximation of the matrix exponential using Schur Decomposition is the same as the MATLAB embedded approximation of the matrix exponential with relative error 10^{-15}.

5 APPLICATIONS

5.1 Linear Differential Equations

The matrix exponential has applications to systems of linear differential equations. The following applications are given in [10]. Recall that a differential equation of the form

$$y' = Cy$$

has solution $e^{Cx}y(0)$. If we consider the vector

$$\mathbf{y}(x) = \begin{pmatrix} y_1(x) \\ \vdots \\ y_n(x) \end{pmatrix},$$

we can express a system of coupled linear differential equations as

$$\mathbf{y}'(x) = A\mathbf{y}(x) + \mathbf{b}.$$

If we make an equation that takes into consideration boundary conditions and use an integrating factor of e^{-Ax} and multiply throughout, we obtain

$$\begin{aligned} e^{-Ax}\mathbf{y}'(x) - e^{-Ax}A\mathbf{y} &= e^{-Ax}\mathbf{b} \\ D(e^{-Ax}\mathbf{y}) &= e^{-Ax}\mathbf{b}. \end{aligned}$$

If we can calculate e^{Ax}, then we can obtain the solution to the system.

5.1.1 Example (homogeneous)

Say we have the system

$$\left\{ \begin{array}{rcrrr} x' & = & 2x & -y & +z \\ y' & = & & 3y & -z \\ z' & = & 2x & +y & +3z. \end{array} \right.$$

We have the associated matrix

$$M = \begin{bmatrix} 2 & -1 & 1 \\ 0 & 3 & -1 \\ 2 & 1 & 3 \end{bmatrix}.$$

In the example above, we have calculated the matrix exponential

$$e^{tM} = \begin{bmatrix} 2e^t - 2te^{2t} & -2te^{2t} & 0 \\ -2e^t + 2(t+1)e^{2t} & 2(t+1)e^{2t} & 0 \\ 2te^{2t} & 2te^{2t} & 2e^t \end{bmatrix},$$

so, the general solution of the system is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = C_1 \begin{bmatrix} 2e^t - 2te^{2t} \\ -2e^t + 2(t+1)e^{2t} \\ 2te^{2t} \end{bmatrix} + C_2 \begin{bmatrix} -2te^{2t} \\ 2(t+1)e^{2t} \\ 2te^{2t} \end{bmatrix} + C_3 \begin{bmatrix} 0 \\ 0 \\ 2e^t \end{bmatrix}.$$

That is,

$$\begin{aligned} x &= C_1(2e^t - 2te^{2t}) + C_2(-2te^{2t}) \\ y &= C_1(-2e^t + 2(t+1)e^{2t}) + C_2(2(t+1)e^{2t}) \\ z &= (C_1 + C_2)(2te^{2t}) + 2C_3e^t. \end{aligned}$$

5.2 Inhomogeneous case - variation of parameters

For the inhomogeneous case, we can use a method akin to variation of parameters. We seek a particular solution of the form $\mathbf{y}_p(t) = e^{tA}\mathbf{z}(t)$:

$$\begin{aligned} \mathbf{y}_p' &= (e^{tA})'\mathbf{z}(t) + e^{tA}\mathbf{z}'(t) \\ &= Ae^{tA}\mathbf{z}(t) + e^{tA}\mathbf{z}'(t) \\ &= A\mathbf{y}_p(t) + e^{tA}\mathbf{z}'(t). \end{aligned}$$

For $\mathbf{y}_p$ to be a solution:

$$\begin{aligned} e^{tA}\mathbf{z}'(t) &= \mathbf{b}(t) \\ \mathbf{z}'(t) &= (e^{tA})^{-1}\mathbf{b}(t) \\ \mathbf{z}(t) &= \int_0^t e^{-uA}\mathbf{b}(u)\,du + \mathbf{c}. \end{aligned}$$

So,

$$\begin{aligned}\mathbf{y}_p &= e^{tA}\int_0^t e^{-uA}\mathbf{b}(u)\,du + e^{tA}\mathbf{c}\\ &= \int_0^t e^{(t-u)A}\mathbf{b}(u)\,du + e^{tA}\mathbf{c},\end{aligned}$$

where $\mathbf{c}$ is determined by the initial conditions of the problem.

5.2.1 Example (inhomogeneous)

Say we have the system

$$\begin{cases} x' &= 2x & -y & +z & +e^{2t} \\ y' &= & 3y & -1z & \\ z' &= 2x & +y & +3z & +e^{2t}. \end{cases}$$

So, we then have

$$M = \begin{bmatrix} 2 & -1 & 1 \\ 0 & 3 & -1 \\ 2 & 1 & 3 \end{bmatrix},$$

and

$$\mathbf{b} = e^{2t}\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}.$$

From before, we have the general solution to the homogeneous equation. Since the sum of the homogeneous and particular solutions give the general solution to the inhomogeneous problem, now we only need to find the particular solution (via variation of parameters).

We have,

$$\mathbf{y}_p = e^t\int_0^t e^{(-u)A}\begin{bmatrix} e^{2u} \\ 0 \\ e^{2u} \end{bmatrix} du + e^{tA}\mathbf{c},$$

$$\mathbf{y}_p = e^t\int_0^t \begin{bmatrix} 2e^u - 2ue^{2u} & -2ue^{2u} & 0 \\ -2e^u + 2(u+1)e^{2u} & 2(u+1)e^{2u} & 0 \\ 2ue^{2u} & 2ue^{2u} & 2e^u \end{bmatrix}\begin{bmatrix} e^{2u} \\ 0 \\ e^{2u} \end{bmatrix} du + e^{tA}\mathbf{c},$$

$$\mathbf{y}_p = e^t \int_0^t \begin{bmatrix} e^{2u}(2e^u - 2ue^{2u}) \\ \\ e^{2u}(-2e^u + 2(1+u)e^{2u}) \\ \\ 2e^{3u} + 2ue^{4u} \end{bmatrix} + e^{tA}\mathbf{c},$$

and

$$\mathbf{y}_p = e^t \begin{bmatrix} -\frac{1}{24}e^{3t}(3e^t(4t-1)-16) \\ \\ \frac{1}{24}e^{3t}(3e^t(4t+4)-16) \\ \\ \frac{1}{24}e^{3t}(3e^t(4t-1)-16) \end{bmatrix} + \begin{bmatrix} 2e^t - 2te^{2t} & -2te^{2t} & 0 \\ \\ -2e^t + 2(t+1)e^{2t} & 2(t+1)e^{2t} & 0 \\ \\ 2te^{2t} & 2te^{2t} & 2e^t \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix},$$

which can be further simplified to get the requisite particular solution determined through variation of parameters.

References

[1] S. Athloen and R. McLaughlin, *Gauss-Jordan Reduction: A Brief History*, American Mathematical Monthly 94, 130-142, 1987.

[2] T.A. Bickart, *Matrix Exponential: Approximation by the Truncated Power Series*, Proc IEEE 56, 372-373, 1968.

[3] P. Davies and N.J. Higham, *A Schur-Parlett Algorithm for computing Matrix Functions*, SIAM J. Matrix Anal. Appl. 25(2), 464-485, 2003.

[4] W. Everling, *On the Evaluation of* e^{At} *by Power Series*, Proc IEEE 54, 413, 1967.

[5] G.H. Gould and C.F. Van Loan,*Matrix Computations*, Johns Hopkins University Press, 1996.

[6] W.A. Harris, J.P. Fillmore, and D.R. Smith, *Matrix Exponential Another Approach*, SIAM Review 43(4), 694-706, 2001.

[7] N.J. Higham, *Functions of Matrices*, Chapter 11 in Handbook of Linear Algebra, Chapman/CRC Press, Boca Raton, 2007.

[8] N.J. Higham, *The Scaling and Squaring Method for the Matrix Exponential Revisited*, SIAM J. Matrix Anal. Appl. 26(4), 1179-1193, 2005.

[9] R.A. Horn, C.R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985.

[10] R.A. Horn, C.R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, 1991.

[11] S.H. Hou, E. Edwin, and W.K. Pang, *On the Matrix Exponential*, International Journal of Mathematics Education in Science and Technology, 37:1,65-70, 2006.

[12] P. Lancaster, M. Tismenetsky, *The Theory of Matrices Second Edition with Applications*, Academic Press, 1985.

[13] D.C. Lay, *Linear Algebra and Its Applications-3rd edition*, Pearson Education, Inc., 2003.

[14] S.J. Leon, *Linear Algebra with Applications-6th edition*, Prentice-Hall,Inc., 2002.

[15] M.L. Liou, *A Novel Method of Evaluating Transient Response*, Proc. IEEE 57, 20-23, 1966.

[16] C. Moler and C. Van Loan. *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*, SIAM Review 45, 3-49 ,2003.

[17] P.N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.

[18] B.A. Schmitt, *An Algebraic Approximation for the Matrix Exponential in Singularly Perturbed Boundary Value Problems*, SIAM J. Numer. Anal. 27(1), 51-66, 1990.

[19] A. Tucker, *The Growing Importance of Linear Algebra in Undergraduate Mathematics*, The College Mathematics Journal, 24, 3-9, 1993.

[20] R.C. Ward, *Numerical Computation of the Matrix Exponential with Accuracy Estimate*, SIAM J. Numer. Anal. 14(4), 600-610, 1977.